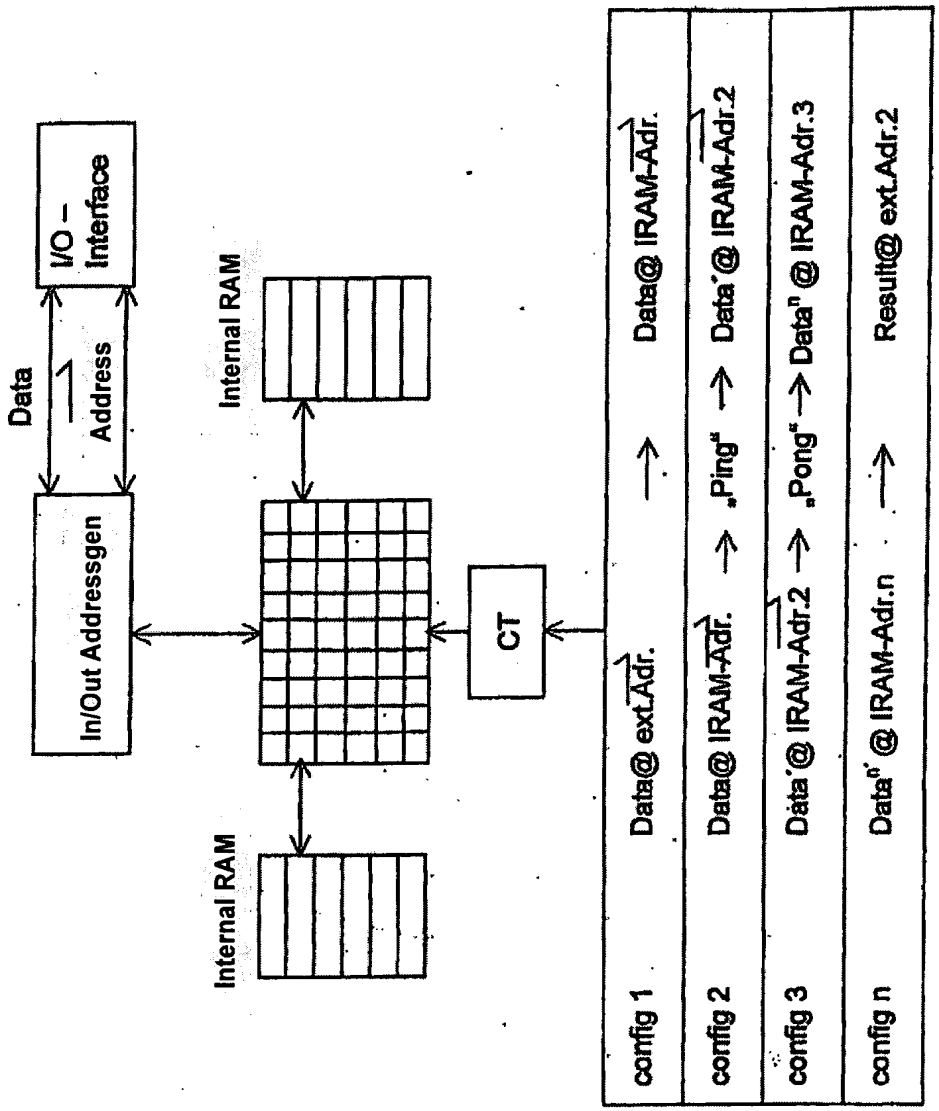

I/O – Interface
Data
Address
In/Out Addressgen
Internal RAM
Internal RAM
CT
config 1
Data@ extAdr.
Data@ IRAM-Adr.
config 2
Data@ IRAM-Adr.
„Ping"
Data'@ IRAM-Adr.2
config 3
Data'@ IRAM-Adr.2
„Pong"
Data''@ IRAM-Adr.3
config n
Data''@ IRAM-Adr.n
Result@ ext.Adr.2
(Fig. 1b I, 1b II, 1b III)
(Fig. 1c)
(Fig. 1d)
(Fig. 1f1, 1f2)

Fig. 1a

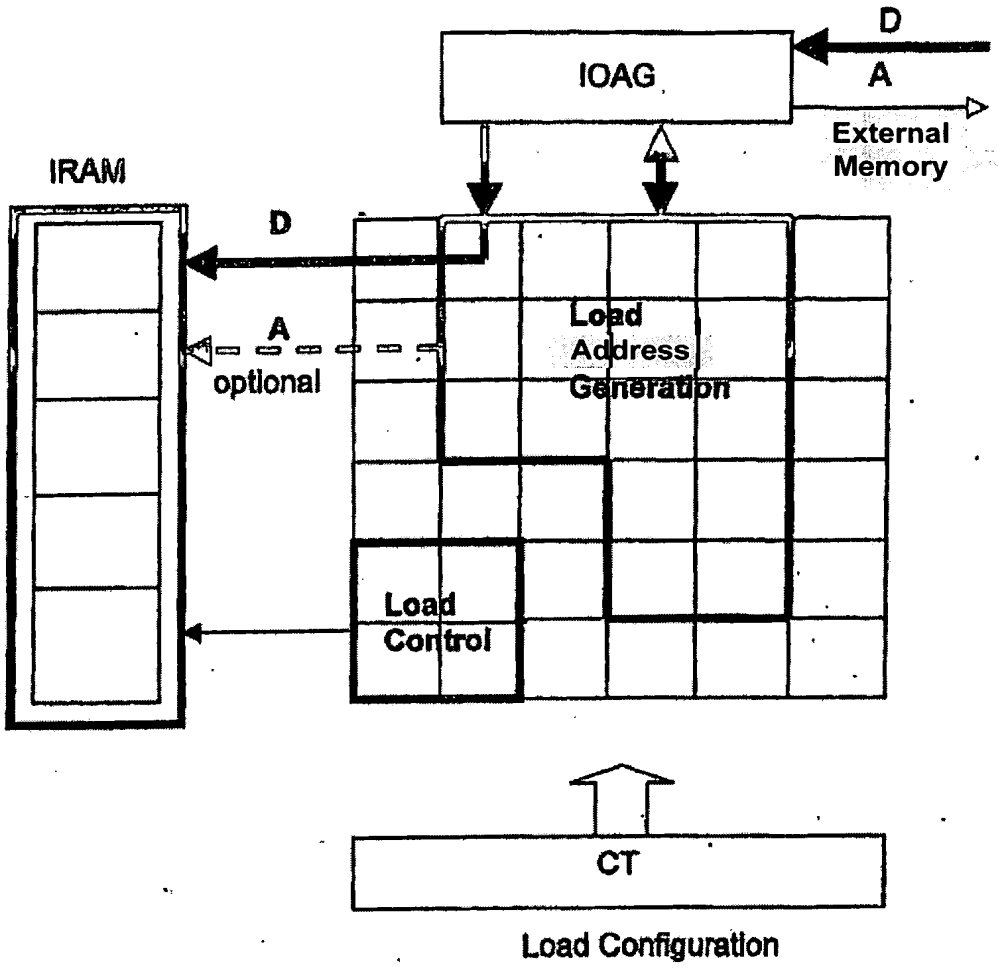

IOAG
D
A
External Memory
IRAM
D
A
optional
Load Address Generation
Load Control
CT
Load Configuration

Fig. 1b I

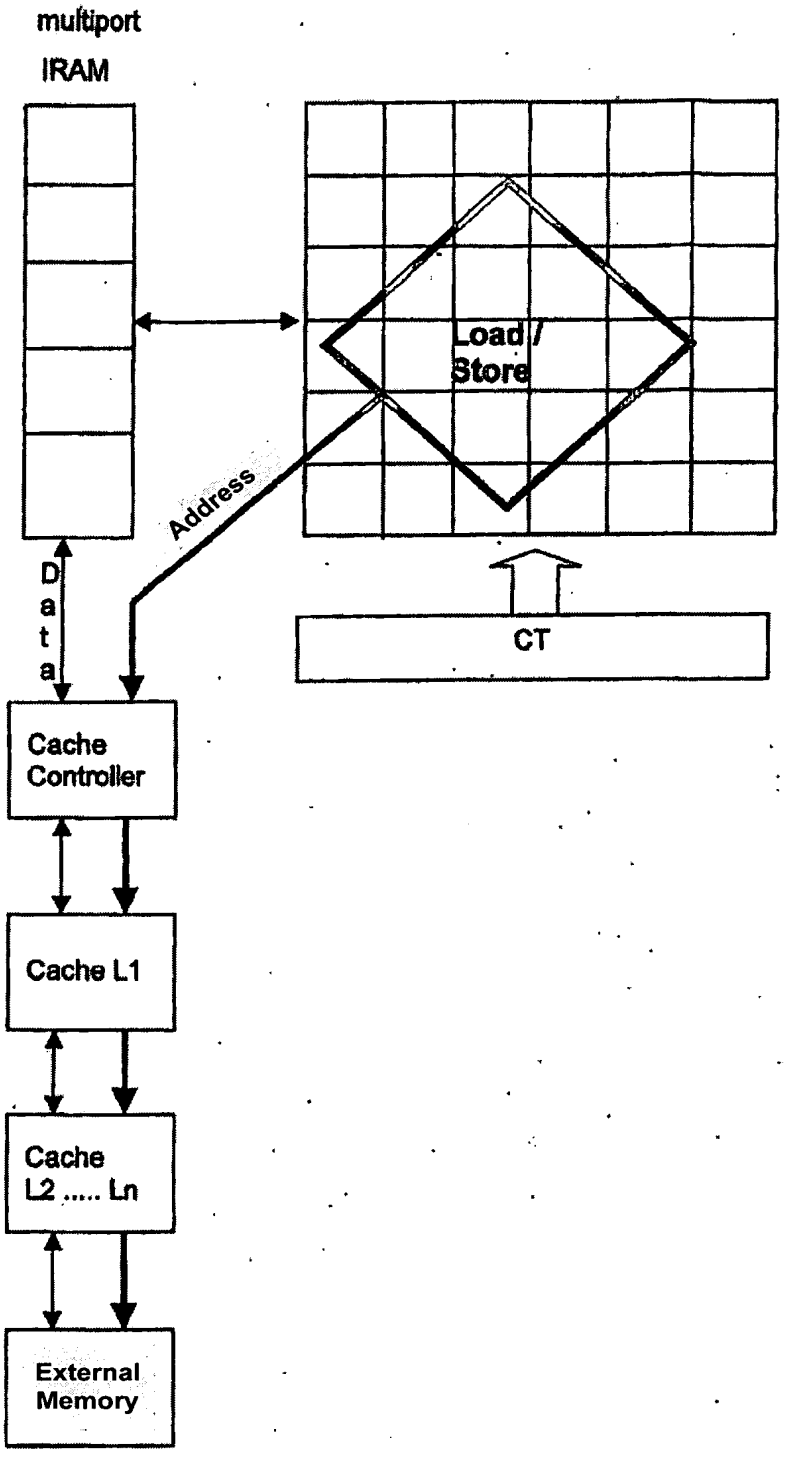

multiport
IRAM
Load /
Store
Address
Data
CT
Cache
Controller
Cache L1
Cache
L2 ..... Ln
External
Memory

Fig. 1b II

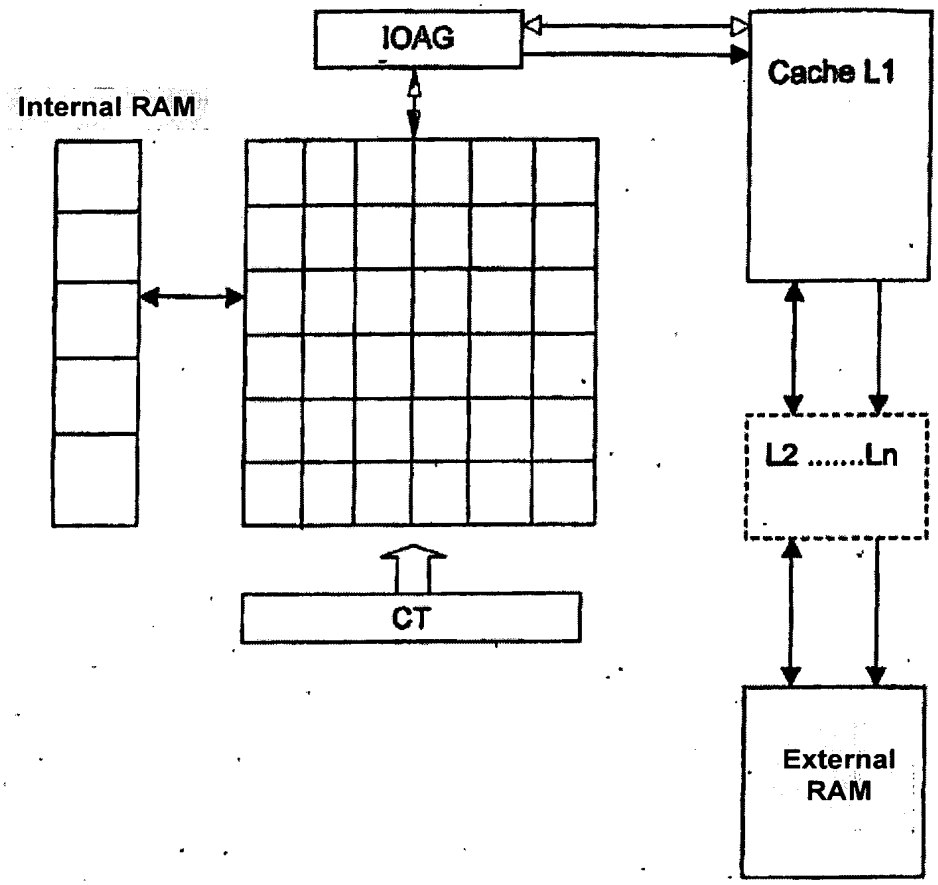


Fig. 1b III

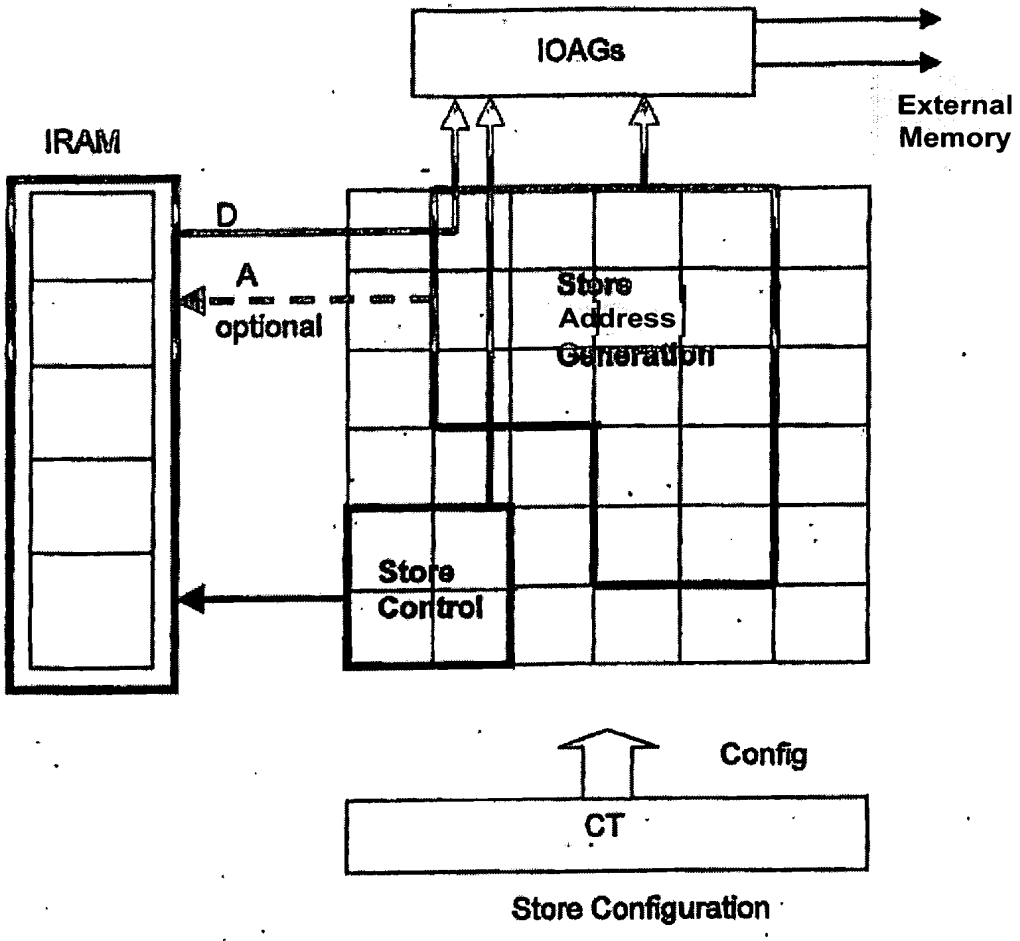

IOAGs
External Memory
IRAM
D
A
optional
Store Address Generation
Store Control
Config
CT
Store Configuration

Fig. 1f1

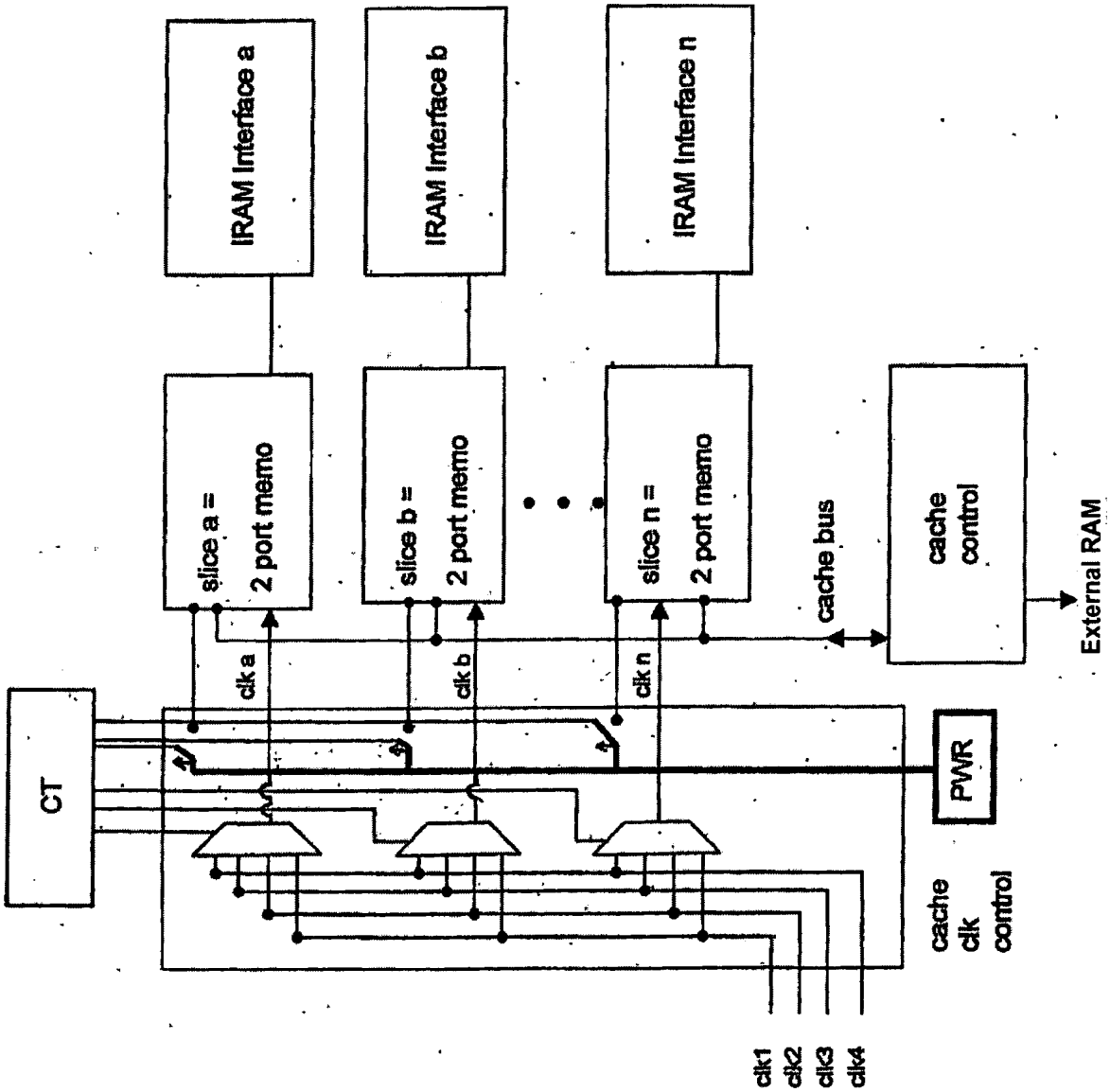

IRAM Interface a
IRAM Interface b
IRAM Interface n
slice a = 2 port memo
slice b = 2 port memo
slice n = 2 port memo
cache bus
cache control
External RAM
clk a
clk b
clk n
CT
PWR
cache clk control
clk1
clk2
clk3
clk4

Fig. 2

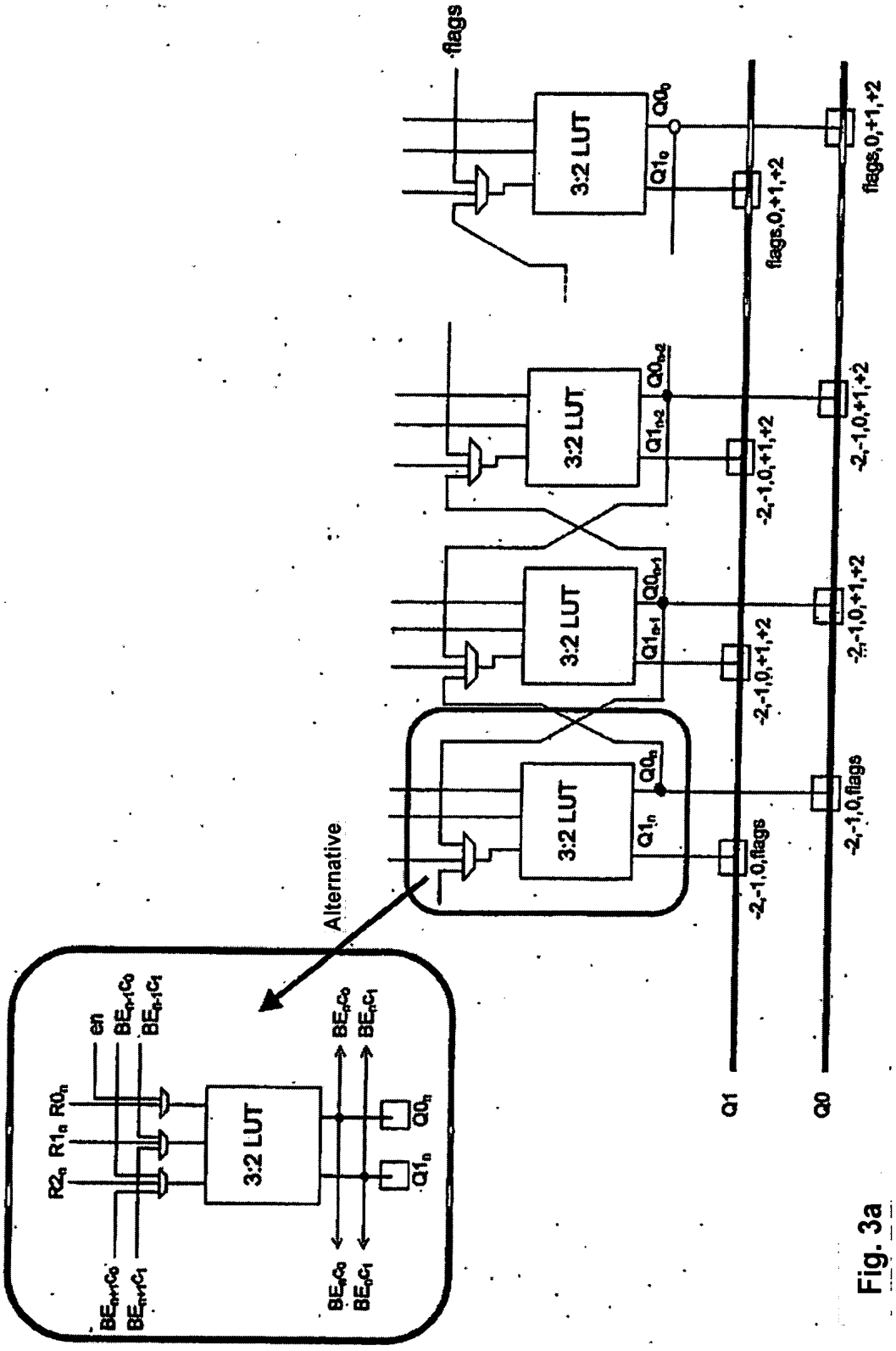
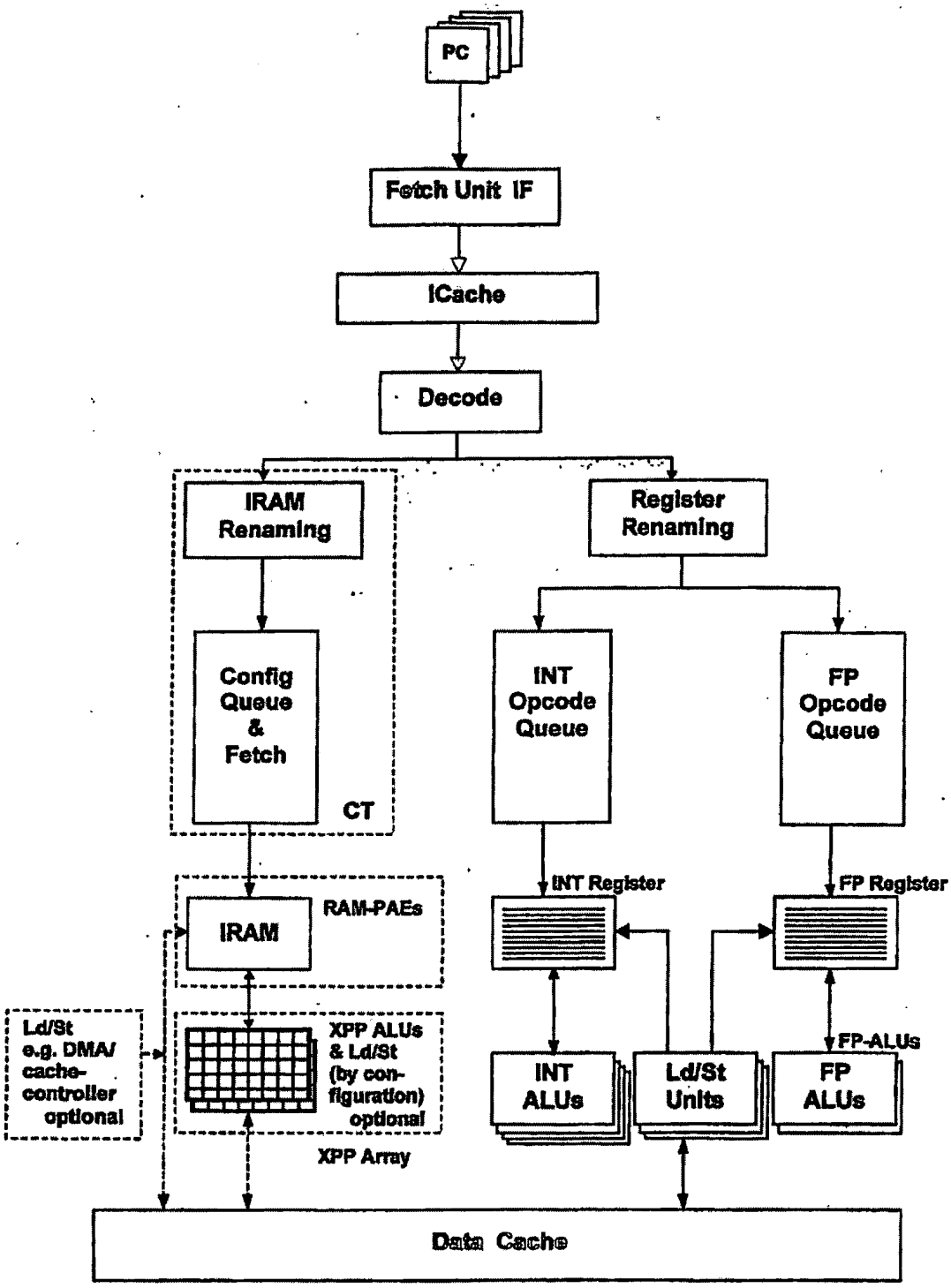
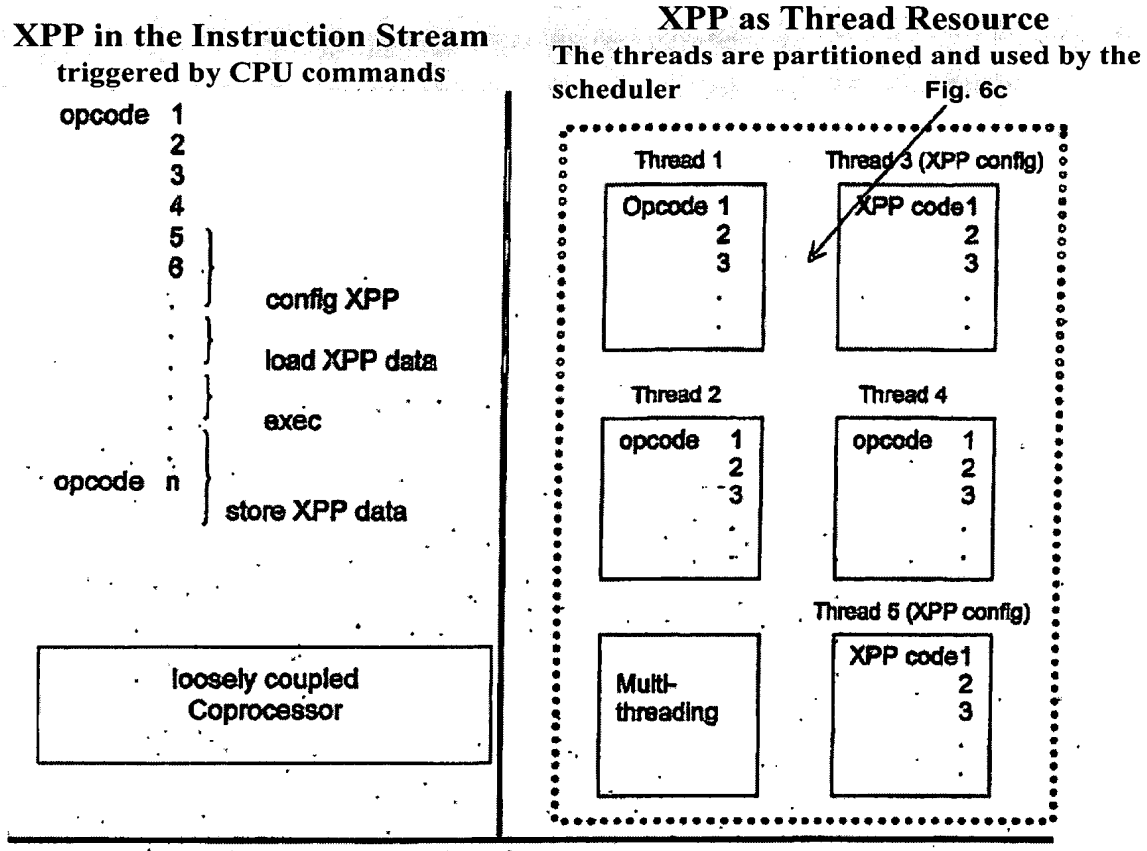

flags
3:2 LUT
Alternative
cin
R2n R1n R0n
BEn+1C0 BEn+1C1
BEnC0 BEnC1
Q1n Q0n
Q1 Q0
flags,0,+1,+2
-2,-1,0,+1,+2
-2,-1,0,flags

Fig. 3a



Possible Structure of an SMT Processor having XPP Thread Resource

Fig. 4a


XPP in the Instruction Stream
triggered by CPU commands
opcode 1
2
3
4
5
6
config XPP
load XPP data
exec
opcode n
store XPP data
loosely coupled
Coprocessor
XPP as Thread Resource
The threads are partitioned and used by the scheduler
Fig. 6c
Thread 1
Opcode 1
2
3
Thread 3 (XPP config)
XPP code1
2
3
Thread 2
opcode 1
2
3
Thread 4
opcode 1
2
3
Thread 5 (XPP config)
Multi-threading
XPP code1
2
3

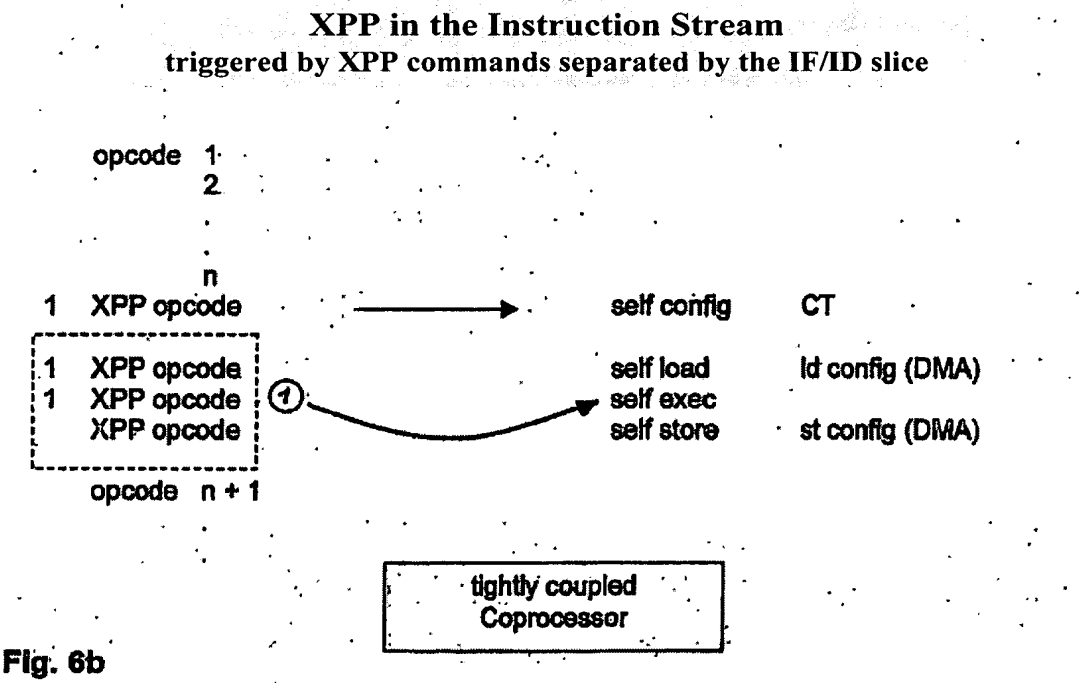

XPP in the Instruction Stream
triggered by XPP commands separated by the IF/ID slice
opcode 1
2
n
1 XPP opcode
1 XPP opcode
1 XPP opcode
XPP opcode
opcode n + 1
self config
CT
self load
ld config (DMA)
self exec
self store
st config (DMA)
tightly coupled
Coprocessor

Fig. 6b